

# Spam Filtering Dengan Metode Pos Tagger Dan Klasifikasi Naïve Bayes

Wirawan Nathaniel Chandra<sup>1</sup>, Gede Indrawan<sup>2</sup>, I Nyoman Sukajaya<sup>3</sup>

<sup>1,2,3</sup>Program Studi Ilmu Komputer, Program Pascasarjana

Universitas Pendidikan Ganesha

Singaraja, Indonesia

e-mail: <sup>1</sup>w1r4\_84@yahoo.com, <sup>2</sup>gede.indrawan@gmail.com, <sup>3</sup>nyoman.sukajaya@undiksha.ac.id

**ABSTRAK.** Email telah menjadi salah satu hal yang penting dalam perkembangan sarana komunikasi. Email telah menggantikan komunikasi yang dilakukan melalui surat. Fasilitas email yang mudah digunakan dan murah mengakibatkan banyaknya email yang berisi iklan dan promosi bisnis masuk ke dalam inbox pengguna email. Email iklan inilah yang disebut sebagai spam mail. Untuk mencegah spam email, maka banyak peneliti yang berusaha untuk melakukan filter terhadap email spam ini secara otomatis, salah satunya adalah dengan menggunakan klasifikasi naïve bayes. Pada penelitian ini selain menggunakan naïve bayes, peneliti juga akan menggunakan metode *POS Tagger*. Diharapkan dengan penambahan metode ini, dapat meningkatkan tingkat akurasi dari klasifikasi naïve bayes.

**Kata kunci:** *spam, spam filter, klasifikasi naive bayes, email spam*

## 1. PENDAHULUAN

Perkembangan dunia teknologi saat ini sudah semakin pesat, begitu pula halnya dengan dunia maya. Email telah menggantikan komunikasi yang dilakukan melalui surat. Email merupakan teknologi yang telah tersedia di internet untuk melakukan komunikasi jarak jauh, dengan menggunakan email, seseorang dapat berkomunikasi dengan satu orang atau ke beberapa orang sekaligus. Email juga dapat melakukan pengiriman file. Salah satu kelebihan email dibandingkan surat adalah waktu yang diperlukan untuk mengirimkan email lebih cepat dibandingkan surat. Waktu yang diperlukan untuk pengiriman surat bergantung pada jarak dari lokasi pengirim ke lokasi penerima.

Kelebihan yang terdapat dalam teknologi email mendorong pertumbuhan pengguna email yang semakin pesat. Teknologi email banyak pula dimanfaatkan bukan untuk tujuan positif sehingga memicu peningkatan spam email. Spam email dapat didefinisikan sebagai “*unsolicited bulk e-mail*” yaitu e-mail yang dikirimkan kepada ribuan penerima (*recipient*). Spam email biasanya dikirimkan oleh suatu perusahaan untuk mengiklankan suatu produk. Hal ini menyebabkan semakin padatnya *queue* atau antrian dari mail server. Pada survey yang dilakukan oleh (Cranor and LaMacchia 1998), ditemukan bahwa 10% dari email yang diterima oleh suatu perusahaan adalah spam email. Tahun lalu spamcop, yang menyediakan layanan untuk menerima laporan tentang spam, menerima lebih dari 199 juta laporan spam. Pada tahun 2008, diperkirakan terdapat 62 triliun spam yang dikirim di seluruh dunia (McAfee 2008).

Terlepas dari waktu yang terbuang untuk menghapus email spam dari *inbox*, spam juga menyebabkan pemborosan uang bagi *user* yang menggunakan koneksi *dial-up*. Selain itu spam juga membuang *bandwidth* dan dapat menyebabkan penerima di bawah umur mengakses situs-situs yang tidak seharusnya. Banyaknya spam memunculkan kerugian dalam hal memakan banyak sumber daya dan memerlukan waktu untuk menghapusnya. Menyaring spam secara manual sulit dilakukan untuk ukuran dokumen email yang sangat besar. Oleh karena itu diperlukan suatu metode yang dapat mengklasifikasikan spam dan bukan spam (*ham*) secara otomatis. Metode penyaringan secara otomatis ini biasanya disebut dengan *spam filtering*.

*Spam filtering* adalah suatu proses yang memisahkan email berdasarkan kategorinya baik spam maupun bukan spam atau biasa disebut *ham*. Dalam pengklasifikasian email spam, dibutuhkan suatu sistem cerdas yang dapat memilah-milah/mengklasifikasikan email spam dan bukan spam (*ham*) secara baik dan benar. Pada proses pengklasifikasian spam email tersebut dapat diterapkan beberapa metode/algoritma yang dapat membantu sistem dalam melakukan pengklasifikasian. Pada penelitian ini akan digunakan metode *POS Tagger* dan klasifikasi Naïve Bayes. Salah satu alasan penggunaan metode Naïve Bayes ini, adalah karena sifatnya yang *historical* (dapat mengingat kejadian sebelumnya). Hal ini sangat cocok untuk *spam filtering*, mengingat jenis email spam yang bermacam-macam dilihat dari isi email dan headernya.

(Natalius 2011) mencoba mengklasifikasikan dokumen dengan metoda klasifikasi naïve bayes. Penentuan kelas dari dokumen sampel dilakukan dengan menghitung besarnya peluang kata-kata pada dokumen suatu kelas yang muncul pada dokumen sampel yang dianalisis. Jadi, kata-kata dalam dokumen berlaku seperti petunjuk,

dan gabungan dari kata-kata tersebut menghasilkan suatu dokumen. Probabilitas sebuah kelas mengandung suatu dokumen merupakan produk dari probabilitas kata-kata dari dokumen tersebut yang terdapat pada kelas. Setelah dilakukan serangkaian pengujian, ternyata klasifikasi naïve bayes berhasil mengklasifikasikan dokumen-dokumen ke dalam kelas tertentu sesuai dengan yang diharapkan.

(Dewi and Supriyanto 2013) mencoba melakukan klasifikasi teks spam dengan menggunakan algoritma naïve bayes. Selain menggunakan algoritma naïve bayes, peneliti juga menggunakan *word vector creation* (TF-IDF, *term frequency*, *term occurrences*, dan *binary term occurrences*) dan *pruner* (*perceptual* dan *by ranking*). Preproses dataset yang dilakukan terdiri dari proses *tokenize*, *filter stopwords*, dan *stem* yang menggunakan algoritma porter. Setelah dilakukan serangkaian pengujian, dapat ditarik kesimpulan bahwa nilai akurasi tertinggi klasifikasi naïve bayes yang dapat dicapai adalah sebesar 84,40% dengan menggunakan TF-IDF tanpa menerapkan *pruner*.

(Agustina, Matulatan, and Bettiza, n.d.) melakukan penelitian klasifikasi *trending* topik twitter dengan penerapan metode naïve bayes. Jejaring sosial Twitter memiliki ringkasan topik yang disebut *trending* topik, merupakan akumulasi yang berdasarkan pada frekuensi topik pembicaraan yang sama pada kurun waktu tertentu secara *real time*. Beragam topik pembicaraan yang kemudian diklasifikasikan berdasarkan kelompok yang lebih umum dimana beberapa topik pembicaraan dapat digolongkan sebagai topik pembicaraan yang sama atau kategori tertentu. Penggolongan *trending* topik ke dalam kategori tertentu tersebut menerapkan metode Naïve Bayes. Untuk itu penelitian digunakan 5 kategori yang masing-masing memiliki 20 fitur. Metode Naïve Bayes menggunakan klasifikasi kata dari topik pembicaraan yang sama dengan membandingkan setiap fitur yang dimiliki oleh tiap kategori. Hasil yang diperoleh dari klasifikasi *trending* topik menggunakan Metode Naïve Bayes pada 30 data uji sederhana dapat menunjukkan pengklasifikasian *trending* topik. Hasil dari data uji coba menunjukkan hasil kategori Religi 16,67%, Olahraga 36,7%, Berita 6,7%, Televisi & Film 6,7% dan Musik 33,3%.

Sebelum adanya penelitian ini proses klasifikasi semuanya dilakukan secara manual. Sebagai contoh kita ambil penelitian klasifikasi *trending* topik twitter, *hashtag* #PeduliSinabung, #IndonesiaJaya sebelumnya tidak termasuk ke dalam topik tertentu, namun jika *user* ingin mengklasifikasikan atau memasukkan kedua *hashtag* ini ke dalam topik tertentu, maka *user* tersebut harus melakukannya secara manual. Misal *hashtag* #PeduliSinabung diklasifikasikan ke dalam topik berita, dan *hashtag* #IndonesiaJaya diklasifikasikan ke dalam topik olahraga. Hal ini cukup memakan banyak waktu jika *user* ingin mengklasifikasikan data yang jumlahnya ribuan atau bahkan puluhan ribu, untuk itu diperlukan sebuah metode yang dapat mengklasifikasikan secara efektif dan otomatis, yaitu dengan menggunakan teknik klasifikasi Naïve Bayes.

Untuk mempersiapkan data sebelum diklasifikasikan dengan Naïve Bayes, para peneliti sebelumnya hanya melakukan praproses data mentah menggunakan metode tokenisasi, *stop-word removal*, dan *stemming* saja. Selain itu untuk menghilangkan kata-kata umum para peneliti sebelumnya hanya menggunakan metode *stop-word removal* saja, hal ini menurut peneliti kurang tepat karena metode ini hanya berdasarkan pada daftar *stop-word* saja. Dimana daftar *stop-word* kadang kala kurang lengkap, yang menyebabkan ada beberapa kata umum yang tidak terfilter. Untuk itu peneliti mencoba menambahkan beberapa metode praproses yaitu: normalisasi email (menghilangkan tanda baca), dan *casefolding* (mengubah semua kata menjadi huruf kecil). Selain itu juga akan ditambahkan proses ekstraksi fitur yang menggunakan *POS Tagger*. Dengan penambahan beberapa metode praproses dan ekstraksi fitur ini diharapkan dapat meningkatkan tingkat akurasi dari klasifikasi Naïve Bayes. Sebab dengan adanya *POS Tagger*, jenis kata yang diklasifikasi menjadi lebih fokus, karena kata yang diambil hanya kata sifat (*adjective*), kata keterangan (*adverb*), kata benda (*noun*) dan kata kerja (*verb*).

Perumusan masalah penelitian ini adalah:

1. Bagaimana rancangan sistem pengklasifikasian spam email menggunakan metode *POS Tagger* dan klasifikasi Naïve Bayes?
2. Bagaimana akurasi sistem pengklasifikasian spam email menggunakan metode *POS Tagger* dan klasifikasi Naïve Bayes?

Tujuan dari penelitian ini adalah untuk mengetahui rancangan dan meningkatkan akurasi pengklasifikasian spam email dengan menggunakan metode *POS tagger* dan klasifikasi Naïve Bayes.

Penggunaan *Part-Of-Speech Tagger* (*POS Tagger*) pada klasifikasi naïve bayes meningkatkan akurasi klasifikasi email spam yang dilakukan secara otomatis, sehingga memudahkan *user* dalam memverifikasi lebih lanjut email yang masuk apakah termasuk spam atau bukan spam (*ham*), tanpa harus memilah-milah email tersebut secara manual.

Pembatasan masalah pada penelitian ini adalah:

1. *Spam filtering* yang diacu pada penelitian adalah *spam filtering* yang diimplementasikan pada program webmail.
2. Klasifikasi Spam email hanya memperhatikan *header* dan *body* dari email yang dalam bentuk bahasa inggris dan tidak dapat melakukan pengecekan terhadap sebuah *attachment* atau *file*.

## 2. METODE PENELITIAN

Tahapan penelitian dimulai dari pengumpulan data email, ekstraksi dokumen email, praproses, pemilihan dan ekstraksi fitur, dan melakukan pemodelan menggunakan klasifikasi naïve bayes, pengujian, dan pengujian hasil. Blok Sistem yang diusulkan ditunjukkan pada Gambar 1.



Gambar 1. Blok sistem yang diusulkan

### Pengumpulan data email

Data email yang diperoleh dari file *archive* <http://www.untroabled.org/> dan <http://spamassassin.apache.org/> sebelum diproses lebih lanjut akan dideteksi apakah termasuk bahasa inggris atau bahasa indonesia. Jika termasuk bahasa inggris, maka email tersebut akan diproses lebih lanjut ke tahap ekstraksi dokumen email. Ekstraksi dokumen *email* ini dimaksudkan untuk mendapatkan bagian *email* yang akan dimasukkan dalam praproses. Dokumen email yang akan diekstraksi hanya memperhatikan *header* dan *body* (dapat berupa *plain text* ataupun *html text*) dari email yang dalam bentuk bahasa inggris dan tidak dapat melakukan pengecekan terhadap sebuah file attachment

### Praproses

Praproses akan dilakukan pada data email mentah yang diperoleh dari proses download file *archive*. Tujuan praproses ini adalah untuk mempersiapkan data agar dapat digunakan pada proses klasifikasi. Tahap praproses ini meliputi:

- Normalisasi Email, proses ini bertujuan untuk menghilangkan tanda baca (*punctuation*) dan simbol selain *alphabet*. Tujuan dari menghilangkan tanda baca adalah agar tanda baca dan simbol selain *alphabet* tidak masuk menjadi token pada saat proses tokenisasi.
- Tokenisasi, merupakan proses pemotongan sebuah dokumen menjadi bagian-bagian, yang disebut dengan token.
- Casefolding*, merupakan proses penyamaan case dalam sebuah dokumen. Ini dilakukan untuk mempermudah pencarian. Tidak semua dokumen teks konsisten dalam penggunaan huruf kapital. Oleh karena itu peran *case-folding* dibutuhkan dalam mengkonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (biasanya huruf kecil).
- Stop-Word Removal*, merupakan proses penghilangan kata-kata yang tidak berkontribusi banyak pada isi dokumen. Kata-kata yang termasuk *stopword* dihilangkan karena memberikan pengaruh tidak baik dalam pencarian dokumen yang dikehendaki user. *StopWord* dalam penelitian ini menggunakan bahasa inggris.

### Pemilihan dan Ekstraksi Fitur

Setelah melalui proses praproses, data akan diseleksi dengan menggunakan *POS tagger* dan *stemming*. *POS Tagger* merupakan proses memberikan kelas pada kata. Kelas kata yang dipilih adalah kata sifat (*adjective*), kata keterangan (*adverb*), kata benda (*noun*) dan kata kerja (*verb*). *Stemming*, bertujuan mengurangi variasi kata yang memiliki kata dasar sama.

### Pembelajaran dan klasifikasi

Proses selanjutnya adalah proses klasifikasi untuk menentukan apakah email ini spam dan bukan spam (*ham*) akan menggunakan *Naive Bayes Classifier*. *Naive Bayes* atau dikenal umum dengan *Naive Bayes Classifier* (NBC) merupakan salah satu algoritma klasifikasi yang menggunakan statistika. Pengklasifikasian ini menggunakan probabilitas sederhana yang diadopsi dari teorema Bayes (statistik Bayesian), dimana menggunakan asumsi naif atau independen yang kuat. Dalam proses pengklasifikasian teks terdapat dua tahapan, yaitu: tahap pelatihan (*training*) dan tahap pengujian (*testing*). Tahap pelatihan merupakan tahap pelatihan

terhadap sejumlah dokumen contoh, sedangkan tahap pengujian merupakan proses pengklasifikasian dokumen baru dan belum diketahui kategorinya.

Dalam terminologi sederhana, sebuah NBC mengasumsikan bahwa kehadiran (atau ketiadaan) fitur tertentu dari suatu kelas tidak berhubungan dengan kehadiran (atau ketiadaan) fitur lainnya. Sebagai contoh, buah mungkin dianggap apel jika merah, bulat, dan berdiameter sekitar 4 inchi. Bahkan jika fitur ini bergantung satu sama lain atau atas keberadaan fitur lain. Sebuah NBC menganggap bahwa seluruh sifat-sifat berkontribusi mandiri untuk probabilitas bahwa buah ini adalah apel.

Tergantung pada situasi yang tepat dari model probabilitas, NBC dapat dilatih sangat efisien dalam *supervised learning*. Dalam aplikasi praktis, parameter estimasi untuk model NBC menggunakan metode *likelihood* maksimum, dengan kata lain, seseorang dapat bekerja dengan model Naïve Bayes tanpa mempercayai probabilitas Bayesian atau menggunakan metode Bayesian lainnya.

Dibalik desain naifnya dan asumsi yang tampaknya terlalu disederhanakan, NBC telah bekerja cukup baik dalam banyak situasi dunia nyata yang kompleks. Pada tahun 2004, analisis masalah klasifikasi Bayesian telah menunjukkan bahwa ada beberapa alasan teoritis untuk keberhasilan yang tampaknya tidak masuk akal dari NBC (Zhang 2004). Selain itu, perbandingan yang komprehensif dengan metode klasifikasi lainnya pada tahun 2006 menunjukkan bahwa klasifikasi Bayes mengungguli pendekatan terbaru, seperti *boosted tree* atau *random forest* (Caruana and Niculescu-Mizil 2006).

Sebuah keuntungan dari NBC adalah bahwa NBC memerlukan sejumlah kecil data pelatihan untuk mengestimasi parameter (rata-rata dan varian dari variabel) yang diperlukan untuk klasifikasi. Karena variabel diasumsikan independen, hanya varian dari variabel-variabel untuk setiap kelas yang perlu ditentukan dan bukan keseluruhan *covariance matrix*.

### Model probabilistic Naïve Bayes

Model probabilitas untuk classifier adalah model kondisional

$$p(C|F_1, \dots, F_n)$$

terhadap variabel kelas dependen C dengan sejumlah kecil hasil atau kelas, tergantung pada beberapa variabel fitur  $F_1$  sampai  $F_n$ . Masalahnya adalah bahwa jika jumlah fitur n besar atau bila fitur bisa mengambil sejumlah besar nilai, maka membuat sebuah model pada tabel probabilitas adalah tidak mungkin. Oleh karena itu kita mereformulasi model untuk membuatnya lebih fleksibel. Menggunakan teorema Bayes, kita menulis

$$p(C|F_1, \dots, F_n) = \frac{p(C)p(F_1, \dots, F_n|C)}{p(F_1, \dots, F_n)}$$

### Pengujian

Pengujian dilakukan untuk melihat apakah data yang telah menjadi *input* akan diproses sesuai dengan output yang diharapkan. Hal ini juga dilakukan untuk mengevaluasi apakah metode yang diusulkan mampu menjawab tujuan yang telah diusulkan. Proses uji coba akan dilakukan dengan cara:

1. Data email yang diperoleh akan dipakai untuk data training, dan data testing. Kedua data ini akan diklasifikasikan ke dalam kelas (spam atau bukan spam).
2. Metode yang diusulkan akan dipakai untuk melatih data-data training dan data-data testing.
3. Untuk mengukur tingkat efisiensi dari metode yang diusulkan akan digunakan *precision* dan *recall*, dimana rumusnya adalah:

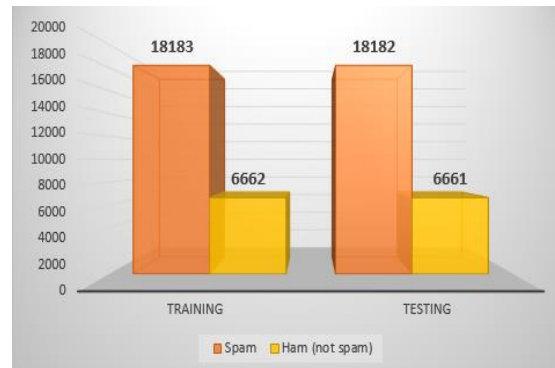
$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

### 3.HASIL DAN PEMBAHASAN

#### PERCOBAAN 1

Ujicoba pertama dilakukan dengan memilih secara acak data latih dan data uji dari 49.688 data email, 2 kategori. Data latih yang dipilih sebanyak 50% dari masing-masing kategori yang ada dan sisanya 50% dijadikan data uji. Jumlah email pada data latih untuk percobaan ini adalah 24.844 email dan data uji sebanyak 24.844 email. Komposisi data latih dan uji percobaan bias dilihat pada Gambar 2.



**Gambar 2.** Komposisi Data Latih dan Uji Percobaan 1

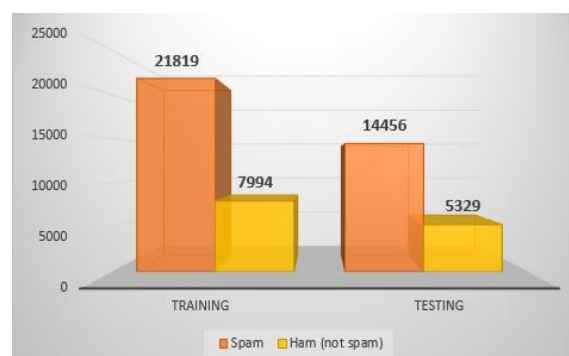
Melalui percobaan ini didapatkan ketepatan prediksi atau akurasi sebesar 78.92% (diprediksi secara tepat sebanyak 19.608 dari 24.844 data email uji), dan error sebesar 21.08% (diprediksi kurang tepat sebanyak 5.236 dari 24.844 data email uji). Tabel 1 berikut ini adalah *confusion matrix* dari ujicoba yang dilakukan:

**Tabel 1.** Confusion Matrix Percobaan 1

Actual/Predicted	Spam	Ham
Spam	13.351	4.831
Ham	405	6.257

#### PERCOBAAN 2

Ujicoba kedua dilakukan dengan memilih secara acak data latih dan data uji dari 49.688 data email, dua kategori. Data latih yang dipilih sebanyak 60% dari masing-masing kategori yang ada dan sisanya 40% dijadikan data uji. Jumlah email pada data latih untuk percobaan ini adalah 29.813 email dan data uji sebanyak 19.875 email. Komposisi data latih dan uji percobaan 2 ditunjukkan pada Gambar 3.



**Gambar 3.** Komposisi Data Latih dan Uji Percobaan 2

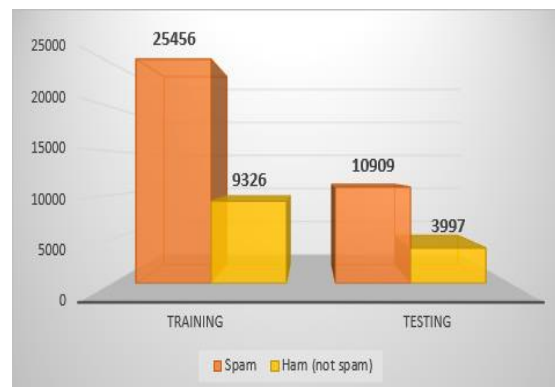
Melalui percobaan ini didapatkan ketepatan prediksi atau akurasi sebesar 83.98% (diprediksi secara tepat sebanyak 16.692 dari 19.875 data email uji), dan error sebesar 16.02% (diprediksi kurang tepat sebanyak 3.183 dari 19.875 data email uji). Tabel 2 berikut ini adalah *confusion matrix* dari ujicoba yang dilakukan:

**Tabel 2.** Confusion Matrix Percobaan 2

Actual/Predicted	Spam	Ham
Spam	11.611	2.935
Ham	248	5.081

### PERCOBAAN 3

Ujicoba ketiga dilakukan dengan memilih secara acak data latih dan data uji dari 49.688 data email, dua kategori, tanpa memperhatikan asal data email yang ada. Data latih yang dipilih merupakan 70% dari masing-masing kategori yang ada dan sisanya 30% dijadikan data uji, dengan harapan persebaran email untuk masing-masing kategori merata. Jumlah data email pada data latih untuk percobaan ini adalah 34.782 email dan data uji sebanyak 14.906 email. Komposisi data latih dan uji percobaan 3 digambarkan pada Gambar 4.



**Gambar 4.** Komposisi Data Latih dan Uji Percobaan 3

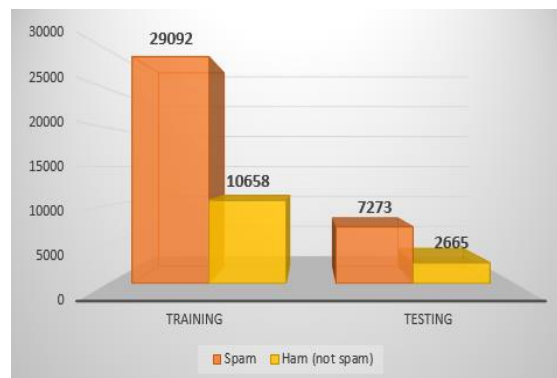
Tabel 3 merupakan *confusion matrix* untuk menunjukkan seberapa banyak ketepatan kategori prediksi yang sama dengan kategori aktual dengan menggunakan Naive Bayes dengan komposisi data latih dan uji yang ada pada percobaan ini. Pada percobaan ini kategori yang tepat diprediksi adalah 12.259 email (Akurasi=82.24%) dan kurang tepat sebanyak 2.647 email (error=17.76%).

**Tabel 3.** Confusion Matrix Percobaan 3

Actual/Predicted	Spam	Ham
Spam	8.452	2.457
Ham	190	3.807

### PERCOBAAN 4

Ujicoba keempat dilakukan dengan memilih secara acak data latih dan data uji dari 49.688 data email, dua kategori. Data latih yang dipilih sebanyak 80% dari masing-masing kategori yang ada dan sisanya 20% dijadikan data uji. Jumlah email pada data latih untuk percobaan ini adalah 39.750 email dan data uji sebanyak 9.938 email. Komposisi pembagian data uji dan data latih untuk percobaan ini adalah seperti pada Gambar 5.



**Gambar 5.** Komposisi Data Latih dan Uji Percobaan 4

Tabel 4 merupakan confusion matrix hasil ujicoba klasifikasi Percobaan keempat dengan Naive Bayes. Matrix tersebut menunjukkan seberapa tepat atau akurat dari klasifikasi email dengan Naive Bayes di mana kategori aktual dari suatu email ditebak dengan tepat kategori aktualnya (kategori prediksi) saat pengujian.

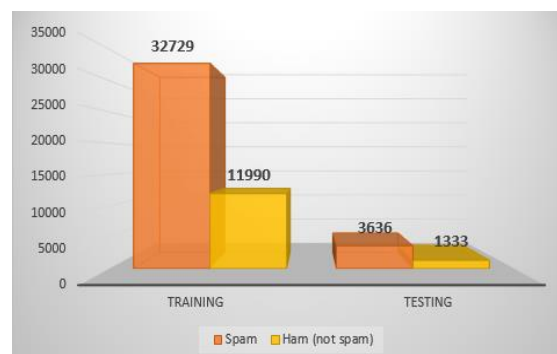
**Tabel 4.** Confusion Matrix Percobaan 4

Actual/Predicted	Spam	Ham
Spam	5.279	1.994
Ham	121	2.544

Akurasi dari ujicoba klasifikasi keempat adalah sebesar 78.72% (diprediksi dengan tepat sejumlah 7.823 dari 9.938 data uji, kategori aktual sama dengan kategori prediksi oleh Naive Bayes) dan error sebesar 21.28% (prediksi kurang tepat sejumlah 2.115 dari 9.938 data uji, kategori aktual berbeda dengan kategori prediksi).

## PERCOBAAN 5

Ujicoba kelima dilakukan dengan memilih secara acak data latih dan data uji dari 49.688 data email, dua kategori. Data latih yang dipilih sebanyak 90% dari masing-masing kategori yang ada dan sisanya 10% dijadikan data uji. Jumlah data email untuk data latih untuk percobaan ini adalah 44.719 email dan data uji sebanyak 4.969 email. Komposisi data latih dan uji percobaan 5 ditunjukkan pada Gambar 6.



**Gambar 6.** Komposisi Data Latih dan Uji Percobaan 5

Melalui percobaan ini didapatkan ketepatan prediksi atau akurasi sebesar 84.30% (diprediksi secara tepat sebanyak 4.189 dari 4.969 data email uji), dan error sebesar 15.69% (diprediksi kurang tepat sebanyak 780 dari 4.969 data email uji). Tabel 5 berikut ini adalah *confusion matrix* dari ujicoba yang dilakukan:

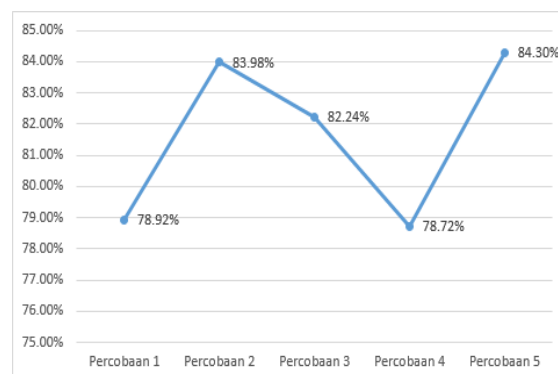


**Tabel 5.** Confusion Matrix Percobaan 5

Actual/Predicted	Spam	Ham
Spam	2.856	780
Ham	0	1.333

#### EVALUASI UJI COBA

Dari kelima ujicoba ini diperoleh beberapa nilai akurasi dari Naive Bayes secara berturut-turut adalah 78.92%, 83.98%, 82.24%, 78.72%, dan 84.30%. Untuk lebih detailnya dapat dilihat pada Gambar 7 di bawah ini.



**Gambar 7.** Hasil Uji Coba

Dari Gambar 7 dapat disimpulkan bahwa nilai akurasi terendah adalah 78.72% dan nilai akurasi tertinggi adalah 84.30% dengan tingkat diagnosa *good classification* (Gorunescu 2011). Melalui pemeriksaan lebih lanjut, beberapa kesalahan pengenalan email *spam* dikarenakan:

1. Email *spam* yang kebetulan memiliki isi seperti *ham*  
Pesan-pesan *ham* pada data uji didominasi oleh pesan dari *mailing list* bertemakan teknologi informasi. Pesan-pesan *spam* yang memiliki tema sangat berbeda seperti obat ataupun judi akan mudah dikenali oleh filter spam, namun jika spam yang dikirim ternyata bertemakan teknologi informasi, kata-kata yang terkandung di dalamnya akan memiliki karakteristik yang mirip dengan mayoritas email *ham*. Akibatnya, filter akan salah mengklasifikasikan email *spam* tersebut sebagai *ham*.
2. Email yang isinya gagal di-decode  
Beberapa email yang menggunakan *character-encoding* maupun *content-transferencoding* khusus gagal di-decode. Kegagalan proses *decode* menyebabkan isi dari email hanya muncul sebagian atau tidak muncul sama sekali sehingga proses klasifikasi didominasi oleh token-token dari *header*.

#### 4.PENUTUP

Pada bagian ini dijabarkan mengenai kesimpulan dari hasil yang diperoleh dalam penelitian ini, dan beberapa saran untuk perbaikan atau pengembangan selanjutnya dari penelitian ini.

##### 4.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dan evaluasi ujicoba, maka didapatkan kesimpulan sebagai berikut:

1. Penggunaan *POS Tagger* pada penelitian ini terbukti mampu memfilter kata-kata yang penting saja, sebab kelas kata yang diambil hanya kata sifat (*adjective*), kata keterangan (*adverb*), kata benda (*noun*) dan kata kerja (*verb*). Pemfokusan kelas kata ini sangat membantu proses klasifikasi Naïve Bayes menjadi lebih efektif.
2. Pada penelitian metode *POS Tagger* dan klasifikasi Naïve Bayes ini hasil akurasi maksimalnya adalah 84.30% untuk data uji email *spam* dan email *ham* dalam bahasa Inggris.



#### 4.2 Saran

Dari beberapa kesimpulan penelitian ini, saran untuk perbaikan dan pengembangan pada penelitian selanjutnya adalah:

1. Perlu dikembangkan *engine* atau metode yang dapat mengenali *character-encoding* maupun *content-transferencoding* khusus, sehingga dapat menghindari email yang gagal di-*decode*.
2. Untuk penelitian selanjutnya dapat dikembangkan *POS Tagger* yang dapat mengenali berbagai macam bahasa, sehingga *spam filtering* tidak terbatas pada Bahasa Inggris saja.

#### DAFTAR PUSTAKA

- [1] Agustina, Puteri Alpita, Tekad Matulatan, and Martaleli Bettiza. n.d. "KLASIFIKASI TRENDING TOPIC TWITTER DENGAN PENERAPAN METODE NAÏVE BAYES."
- [2] Caruana, Rich, and Alexandru Niculescu-Mizil. 2006. "An Empirical Comparison of Supervised Learning Algorithms." *Proceedings of the 23rd International Conference on Machine Learning C* (1): 161–68. doi:10.1145/1143844.1143865.
- [3] Cranor, L. F, and B. A LaMacchia. 1998. "Spam! Communications of the ACM" 41 (8): 74–83.
- [4] Dewi, Ika Novita, and Catur Supriyanto. 2013. "Klasifikasi Teks Pesan Spam Menggunakan Algoritma Naïve Bayes" 2013 (November): 156–60.
- [5] McAfee. 2008. "The Carbon Footprint of Email Spam Report The Carbon Footprint of Email Spam Report." *McAfee, Inc.*
- [6] Natalius, Samuel. 2011. "Metoda Naïve Bayes Classifier Dan Penggunaannya Pada Klasifikasi Dokumen," no. 3.
- [7] Zhang, Harry. 2004. "The Optimality of Naive Bayes." *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference FLAIRS 2004* 1 (2): 1–6. doi:10.1016/j.patrec.2005.12.001.